COBB

Inside MICROSOFT ACCESS

June 1994 • Vol. 2 No. 6

Tips & techniques for Microsoft Access • Windows

Using color to identify the active field on a form

When you browse the data on a form, Access identifies the currently active field by placing the cursor in that field. If you open a field that contains information, that information appears in reverse video. However, if you want to really call attention to the active field, you can do so by adding a splash of color.

In this article, we'll show you how to write a pair of Access basic functions that changes the color of a field's border when you enter that field and returns the border to its original color when you exit the field. Since your form must set the border color at run-time, this function will work only with Access 2.0.

What's involved?

To complete this technique, we first need to build a form. On the form, we'll place text controls for each of the fields we want to include. Then, we'll increase the border width so that the color border will be more prominent. After we finish building the form, we'll determine the color we want to use to highlight active fields.

The real magic happens in an Access Basic module. Fortunately, neither of the functions we'll write includes more than four lines—and Access supplies two of them for you! The first function identifies the currently active field and changes the border color. A companion function resets the color when you leave the field.

To activate the functions, we'll associate them with the On Got Focus and On Lost Focus event properties for each of the text controls on the form. Now, let's build the form we'll use for this technique.

Creating the form

We'll use the Customers table in the NWIND.MDB database as the base table for our form. Begin by opening the database NWIND.MDB. In the Database window, click the Table button and then the Customers table to select it. Then, click the New Form button ( ) on the toolbar to open the New Form dialog box. Click the FormWizards button to start building the form.

In the first Wizard screen, select Single-Column and then click OK. In the next screen, double-click the fields Company Name, Contact Name, Contact Title, and Phone in the Available fields list box to add them to the Field order on form list box. Click Next to open the third Wizard screen.

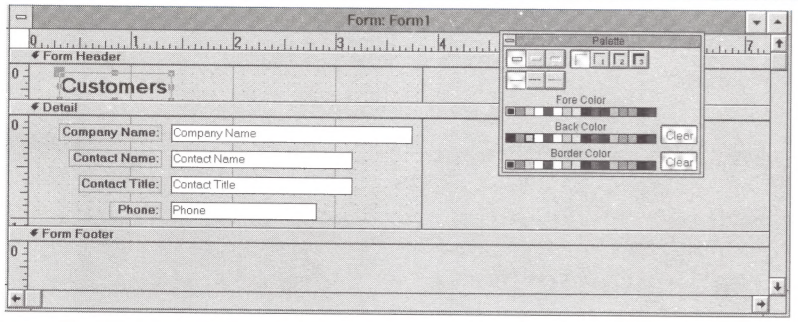
Choose Standard from the form style options and then click Next to move on. Although you can apply this technique to most form styles, you can't accept the default choice, Embossed. Embossed forms

IN THIS ISSUE

use sunken buttons, and Access won't let you change the border width of sunken buttons.

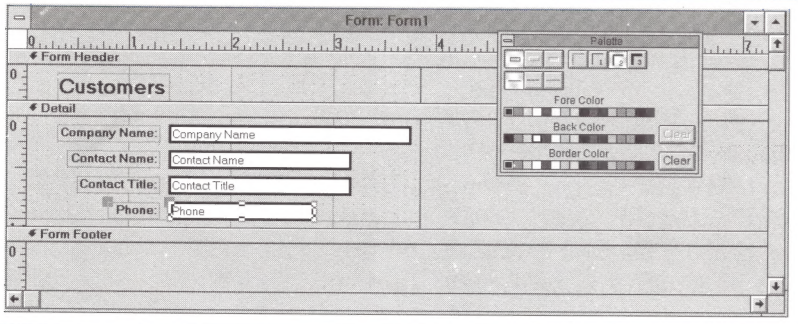
On the final Wizard screen, select the Modify the form's design option and click Finish. The form will open in Design View, as shown in Figure A. Now, you need to prepare the form for the Access Basic functions you'll write later.

Figure A



After you use the Form Wizard to build the form, it opens in Design View.

Figure B



Increase the border width on the text controls to two points.

Preparing the form

If the style and color Palette isn't visible, click on the toolbar's Palette button ( ) to display it. Next, click on the Company Name text control to select it. (The text control appears as a white box with the field name inside it.) Click the 2 point Border Width button ( ) on the Palette to increase the border size.

Select the text controls for Contact Name, Contact Title, and Phone and increase their border size to two points as well. At this point, your form should look like the one shown in Figure B. Now, you're ready to associate the form's event properties with the Access Basic functions.

To display the properties dialog box, you first select the Company Name text control. Then, click on the Properties button ( ) on the toolbar. In the dropdown menu at the top of the properties dialog box, choose Event Properties if it's not already showing.

Next, locate the On Got Focus property. In the On Got Focus text box, type

```
=SetBorderColor()
```

Then, select the On Lost Focus text box and type

```
=ResetBorderColor()
```


Inside MICROSOFT ACCESS™

Inside Microsoft Access (ISSN 1067-8204) is published monthly by The Cobb Group.

Prices
Domestic ..... $59/yr. ($7.00 each)
Outside US ..... $79/yr. ($8.50 each)

Phone
Toll free ..... (800) 223-8720
Local ..... (502) 491-1900
Customer Relations Fax ..... (502) 491-8050
Editorial Department Fax ..... (502) 491-3433

Address
You may address tips, special requests, and other correspondence to
The Editor, *Inside Microsoft Access*
9420 Bunsen Parkway, Suite 300
Louisville, KY 40220

For subscriptions, fulfillment questions, and requests for bulk orders, address your letters to
Customer Relations
9420 Bunsen Parkway, Suite 300
Louisville, KY 40220

Advertising
For information about advertising in Cobb Group journals, contact Tracee Bell Troutt at (800) 223-8720, extension 430.

Postmaster
Second class postage is pending in Louisville, KY. Send address changes to
Inside Microsoft Access
P.O. Box 35160
Louisville, KY 40232

Authorized Canada Post International Publications Mail (Canadian Distribution) Sales Agreement #XXXXXX CANADA GST #123669673. Send returns to Canadian Direct Mailing Sys. Ltd., 920 Mercer Street, Windsor, Ontario, N9A 7C2. Printed in the USA.

Staff
Editor-in-Chief ..... Michael Stephens
Editors ..... Meredith Little
Polly Blakemore
Elizabeth Welch
Production Artists ..... Maureen Spencer
Margueriete Stith
Design ..... Karl Feige
Managing Editor ..... Elayne Noltemeyer
Circulation Manager ..... Marjorie Glassman
Editorial Director ..... Jeff Yocom
Publishers ..... Mark Crane
Jon Pyles

Back Issues
To order back issues, call Customer Relations at (800) 223-8720. Back issues cost $7 each, $8.50 outside the US. You can pay with MasterCard, VISA, Discover, or American Express, or we can bill you. Please identify the issue you want by the month and year it was published. Customer Relations can also provide you with an issue-by-issue listing of all articles that have appeared in *Inside Microsoft Access*.

Finally, type

```
=SetBorderColor()
```

in the On Key Down text box. After you make these changes, your properties dialog box should look like the one shown in Figure C.

The On Got Focus and On Lost Focus event properties tell Access to execute the SetBorderColor() and ResetBorderColor() functions when you enter or exit the Company Name field. The form will use the On Key Down event only when you modify the first field of the record the form displays when it opens. You don't need to associate a function with the On Key Down event for any field except Company Name.

Now, select the Contact Name field and display its event properties. Associate the SetBorderColor() and ResetBorderColor() functions with the On Got Focus and On Lost Focus properties as you did for the Company Name control. Repeat this process for the Contact Title and Phone fields.

Now, choose Save from the File menu. Type *Chameleon* in the Form Name text box and click OK to save the form. When you finish, you just need to create the SetBorderColor() and ResetBorderColor() functions.

Functions follow form

To create these functions, first close the Chameleon form. Then, click the Module button on the Database window. Double-click the Utility Functions module name to open the Utility Functions module.

Now, choose New Procedure... from the Edit menu. In the New Procedure dialog box, select Function from the Type options. Then, type *SetBorderColor* in the Name text box. Click OK to dismiss the dialog box and create the function outline.

Access already displays the first and last lines of our first function. Type

```
On Error Resume Next
Screen.ActiveControl.BorderColor = 255
```

to complete the function, as shown in Figure D. These lines tell Access to set the border color on the active control to the value 255—red. The On Error statement tells Access to continue to the end of the function if it encounters an error.

You must include the On Error statement because Access generates an error when it first opens the form. When a form opens, the first field of the first record receives the focus without becoming the active control. When Access attempts to identify the active control in the SetBorderColor() function, it returns an error. The On Error statement tells Access to continue. Now, let's build the ResetBorderColor() function.

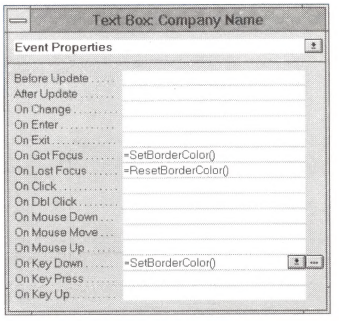
Choose New Procedure... again from the Edit menu. Select Function from the Type options and type *ResetBorderColor* in the Name text box. Click OK to open the function definition. Between the opening and closing lines, type

```
Screen.ActiveControl.BorderColor = 0
```

to reset the border color to black.

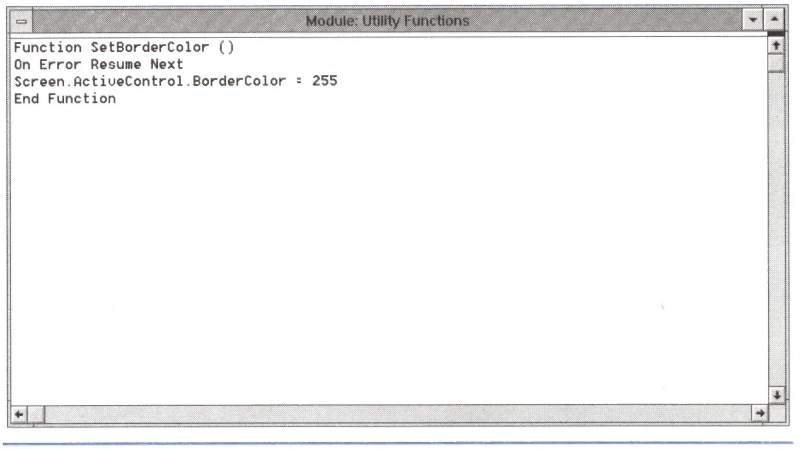
Next, choose Compile Loaded Modules from the Run menu. Then, choose Save from the Edit menu. Close the Module window to return to the Database window. Now, you can use the Chameleon form.

Figure C



Supply the appropriate function names for the On Got Focus, On Lost Focus, and On Key Down event properties.

Figure D



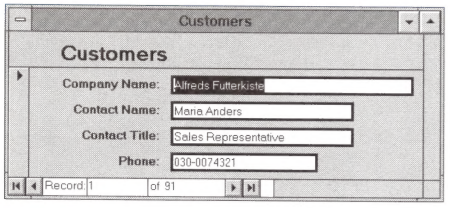
Type the action lines of the function between the opening and ending lines.

Using the form

Click Form on the Database window to display the available forms. Then, double-click Chameleon to open the Chameleon form. As Figure E shows, when the form opens, all the fields have black borders.

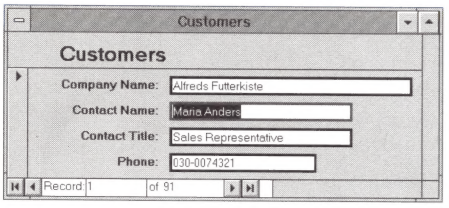
Now, press [Tab] to move through the fields on the form. As Figure F shows, the form changes the color of the border around the active field from black to red. The color resets when you leave the field. When you advance to the next record in the form, Access highlights the active control when it opens that record. The active control will usually be the first or last text box on the form.

Figure E



When the form opens, the Company Name field appears bordered in black.

Figure F



The border around the active field changes from black to red.

Notes

We chose the color red to identify the active fields on our form, but you can choose any color that Access will display on your monitor. Access requires you to identify a color by its numerical value. Fortunately, it's easy to find the value for the color you want.

First, open a form in Design View. Then, display the layout properties for any object on the form. Next, locate the Back Color property. The Back Color text box displays the numerical equivalent of the currently selected back color.

Click the color you want to identify in the Palette's Back Color options. The Back Color text box will display the numerical value for the color you select. Remember to return the Back Color property to its original setting, unless you want to keep the new color value you select in the Palette.

In Access 2.0, you can manipulate almost any property of an Access form at run-time. You need to consult online help and the Access user's guide for specific information about event properties and other form characteristics.

Conclusion

A little color makes it a lot easier to identify the active field in a form. In this article, we showed you how to use a pair of Access Basic functions to change the border color of the active field. ❖

Macro Tip

Filtering records with the Where Condition argument

Microsoft Access macros increase your productivity by automating common tasks. The OpenForm action allows you to create a macro that opens a specific Access form when you execute the macro. However, unless you supply an expression in the OpenForm action's Where Condition argument, the macro will display all the records in the table or query linked to that form.

Fortunately, the Where Condition argument allows you to provide selection criteria that limit the records the form displays. In this article, we'll show you how to build a macro that opens a form and displays only the records you select with the Where Condition argument.

What is where?

When you select an action in an Access macro, the Macro window displays a list of available arguments below the macro grid. The OpenForm action provides the arguments Form Name, View, Filter Name, Where Condition, Data Mode, and Window Mode.

The Form Name argument, as you'd expect, tells the macro which form to open. The View argument tells the macro how to display the records you select. Filter Name

allows you to supply a query name to use when displaying the records for the form.

The last two arguments, Data Mode and Window Mode, govern the form's presentation. You use Data Mode to determine whether you can edit the records the form displays. Window Mode lets you choose the type of form the macro displays.

Unlike other arguments, which require only a name or a selection from the argument's dropdown menu, the Where Condition argument requires you to build an expression. The Where Condition argument supplies the information for the WHERE argument of the SQL statement the macro builds when it executes the OpenForm action.

To build the expression, you use object names in brackets to identify the field names in the expression. When Access opens the form you specify, it uses the Where Condition argument to identify the specific records to display.

When you build the Where Condition expression, you must use a valid field name instead of the control name or label assigned to the control that displays that field on the form. For example, if you build a form that displays the contents of a field named Last Name in a text box control named Customer with the label Customer Last Name, you identify the field in the Where Condition expression as [Last Name]. To demonstrate, let's work through a simple example.

Building an OpenForm macro action

Since we'll need an existing database to demonstrate this technique, begin by opening the database NWIND.MDB in the main Access directory. (If you're using Access 2.0, NWIND.MDB resides in the SAMPAPPS directory in the main Access directory.) We'll build a macro that opens the Order Review form. When the Order Review form opens, it will display, in Datasheet View, only the records associated with sales representative Nancy Davolio.

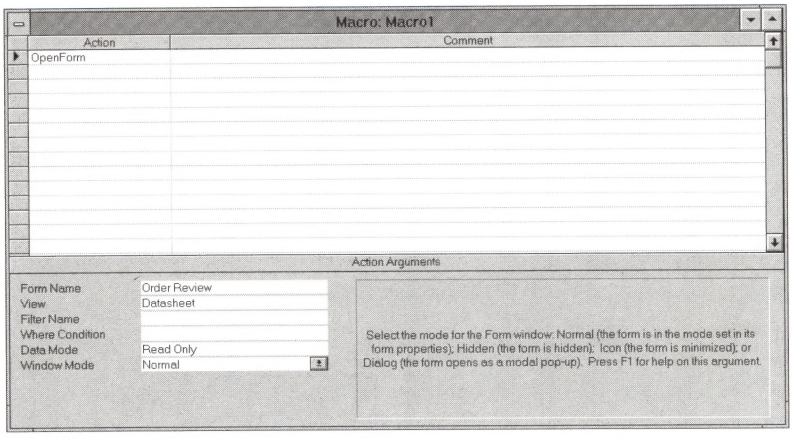
In the Database window, click the Macro button. Then, click New to open a new macro definition. In the Action cell of the first row, type *OpenForm* or choose Open Form from the dropdown menu.

Supplying the arguments

Make the following changes in the Action Arguments section of the Macro window. Type *Order Review* in the Form Name text box. Type *Datasheet* in the View text box. Leave the Filter Name text box empty. Type *Read Only* in the Data Mode text box, and type *Normal* in the Window Mode text box, as shown in Figure A.

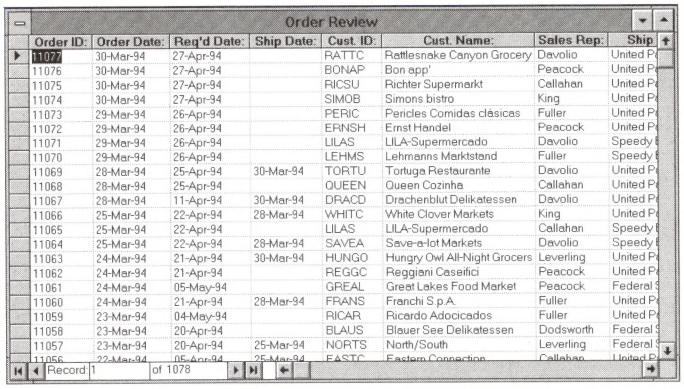
Before we supply an argument for the Where Condition text box, let's look at the records this macro returns. First, you need to save the macro. To do so, choose Save from the File menu. Type *Open Orders* in the Macro Name text box and click OK to dismiss the Save As dialog box. Now, click the Run icon ( ) on the toolbar to execute the macro. As Figure B shows, Access displays the data associated with the Order Review form in Datasheet View.

Figure A



The macro definition window should look like this.

Figure B



Access displays all the records associated with the Order Review form.

Now, let's modify the macro to display only the records associated with sales representative Davolio.

Adding the Where Condition argument

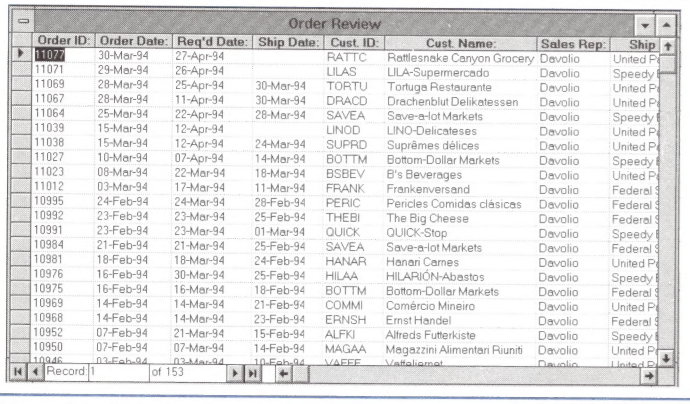
Close the Order Review form. Then, type the expression

```
[Last Name] = "Davolio"
```

into the Where Condition text box. Choose Save from the File menu to save the macro definition. Then, click the Run button on the toolbar to execute the macro. As Figure C shows, the macro selected only the records in which the Sales Rep column displays the name Davolio.

You may have noticed that the form displays the name Davolio in a column titled Sales Rep. In this case, Sales Rep is the label for the text control on the Order Review form. However, that control displays a field named Last Name. The Where Condition argument requires the field name instead of the control's label.

Figure C



The Where Condition argument limits the records on this form to those associated with Nancy Davolio.

Taking the next step

Access lets you associate a macro with a button on a form. You can use a macro like the one we just created to display only the records associated with a particular record of a form. For more information, see the article "Using Object Identifiers to Create Context-Sensitive Macros" below.

Conclusion

Access macros are great timesavers. The OpenForm macro action lets you open a specific form when you execute the macro. You can use the Where Condition argument with the OpenForm macro action to display only selected records when the form opens. ❖

Macro Tip

Using object identifiers to create context-sensitive macros

Relational databases like the ones you create with Microsoft Access allow two or more tables to share information. For example, you can link tables or queries to identify all the orders generated by a particular sales representative. Often, you need to perform the same task many times as you browse a database. For example, you might want to display the individual sales records for each representative you employ.

Although it's not difficult to create a query that returns this type of information, you can create a macro that generates the list you want on the fly. You also can associate that macro with a button on the form you're using to examine your database. In this article, we'll show you how to create a macro that returns only the information related to the record currently displayed in a form.

How it works

In "Filtering Records with the Where Condition Argument" on page 4, we show you how to display only specific records when you execute the OpenForm action in a macro. However, if the records you want to display change as you progress through a database, you also need to update the Where Condition argument when the context changes. Fortunately, Access lets you do this.

Instead of providing the Where Condition argument with the exact information you're trying to locate, you supply an

identifier for the type of information you want to display. For example, if you want to display the records associated with the sales representative in the current record of the form you're examining, you can tell Access to identify the sales representative in that record and then supply that name to the Where Condition argument.

Access lets you use object identifiers to point to the contents of a particular field. When you build the expression for the Where Condition argument in your macro, you enter the object identifier expression instead of a specific value. As you move through the database, the object identifier always points to the information in the active record of the form.

Let's build an example to demonstrate this handy technique. We'll use the Employees form in the NWIND.MDB database to identify the records we want to display in the Order Review form. We'll also create a macro that includes the object identifier expression in the Where Condition argument of the OpenForm action.

Getting started

Before we build our macro, we need to locate the exact information we'll use to create the object identifier expression.

Since we want our macro to display all the records associated with a particular sales representative, we'll need to select a field that identifies the sales representative in both the Employees form and the Order Review form. Both forms contain a text control that displays the Last Name field. We'll use the Last Name field as the object identifier.

To build an object identifier that points to the Last Name field, we work from the most general category to the most specific. First, we know that the object we're identifying resides on a form. We'll use the [Forms] identifier to indicate this.

Next, we know that the specific form we're using is the Employees form. Consequently, the next item in our identifier is the form name [Employees]. Finally, we identify the Last Name field with the expression [Last Name]. You use the specifier symbol (!) to join the individual components. The complete expression looks like

```
[Forms]![Employees]![Last Name]
```

Similarly, we must identify the Last Name field in the Order Review form. However, since our macro makes the Order Review form the active form, we won't need to supply the complete identifier. You can simply use the field name to identify a field in the active form. In this case, we'll use the identifier

```
[Last Name]
```

Now, let's build the macro that we'll use to open the Order Review form.

Building the macro

To build the macro, first open the database NWIND.MDB and click the Macro button in the Database window. Then, click the New button to open a new macro definition. In the Action cell of the first row, type *OpenForm* and press [Enter] or choose OpenForm from the dropdown menu. Next, you supply the arguments for the OpenForm action.

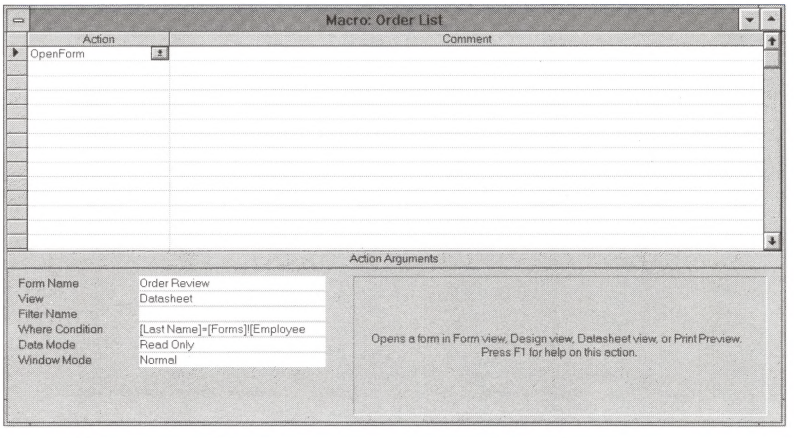
Enter the following information in the Action Arguments section of the Macro window. Type *Order Review* in the Form Name text box. Type *Datasheet* in the View text box. Leave the Filter Name text box empty. In the Where Condition text box, type the expression

```
[Last Name] = [Forms]![Employees]![Last Name]
```

Type *Read Only* in the Data Mode text box and accept the default setting Normal in the Window Mode text box.

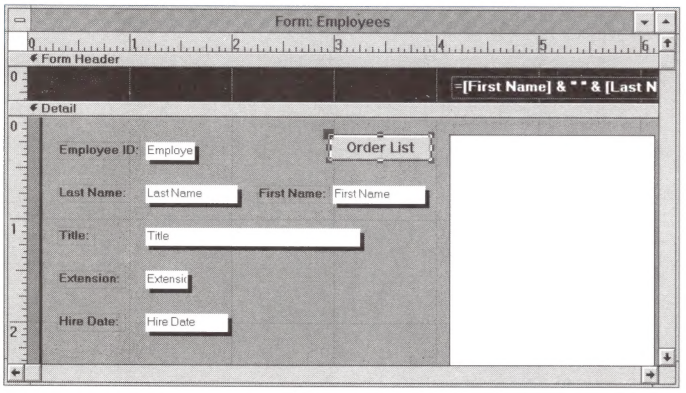
Next, choose Save As... from the File menu. Type *Order List* in the Macro Name

Figure A



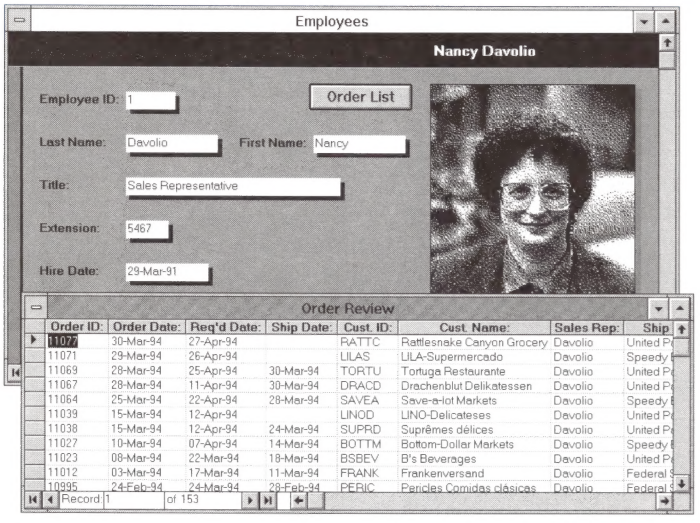
Your completed macro definition should look like this.

Figure B



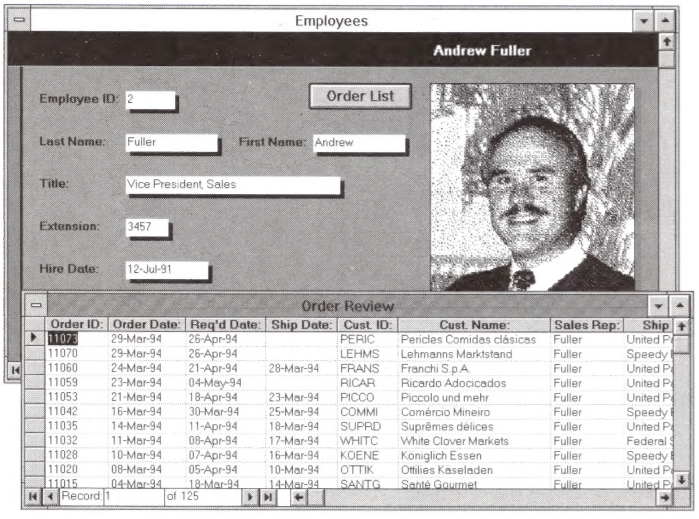
Drag the Order List macro onto the form to create the Order List button.

Figure C



Access displays the orders associated with the current record in the Employees form.

Figure D



Access displays the Order Review records for Andrew Fuller when you select his record in the Employees form.

text box and click OK to dismiss the Save As dialog box. At this point, your macro definition should look like the one shown in Figure A on the previous page. Close the macro window.

Creating the macro button

Drag and drop technology makes it easy to associate a macro with a button on a form. You just create the macro and then drag its name from the macro list in the Database window onto the form. Access automatically creates the button and installs it on the form.

Since we want to activate our macro from the Employees form, begin by clicking the Form button on the Database window. Next, double-click Employees in the form list to open the Employees form. Click the Design View button ( ) on the toolbar to switch to Design View.

Now, click the Show Database Window button ( ) on the toolbar to display the Database window. (If you can see any part of the Database window onscreen, you can simply click on the window to activate it.) Next, click the Macro button to display the list of available macros. Locate the Order List macro and then drag it onto the Employees form.

As Figure B shows, Access creates a button named Order List on the form. When you click this button, it activates the Order List macro. Now, you're ready to use this button to display the Order Review form.

Using the button

To demonstrate the Order List macro, click the Form View button ( ) on the toolbar to activate the form. The form displays the employee information for sales representative Nancy Davolio. Now, click the Order List button. As Figure C shows, Access displays the orders associated with Davolio on the Order Review form in Datasheet View.

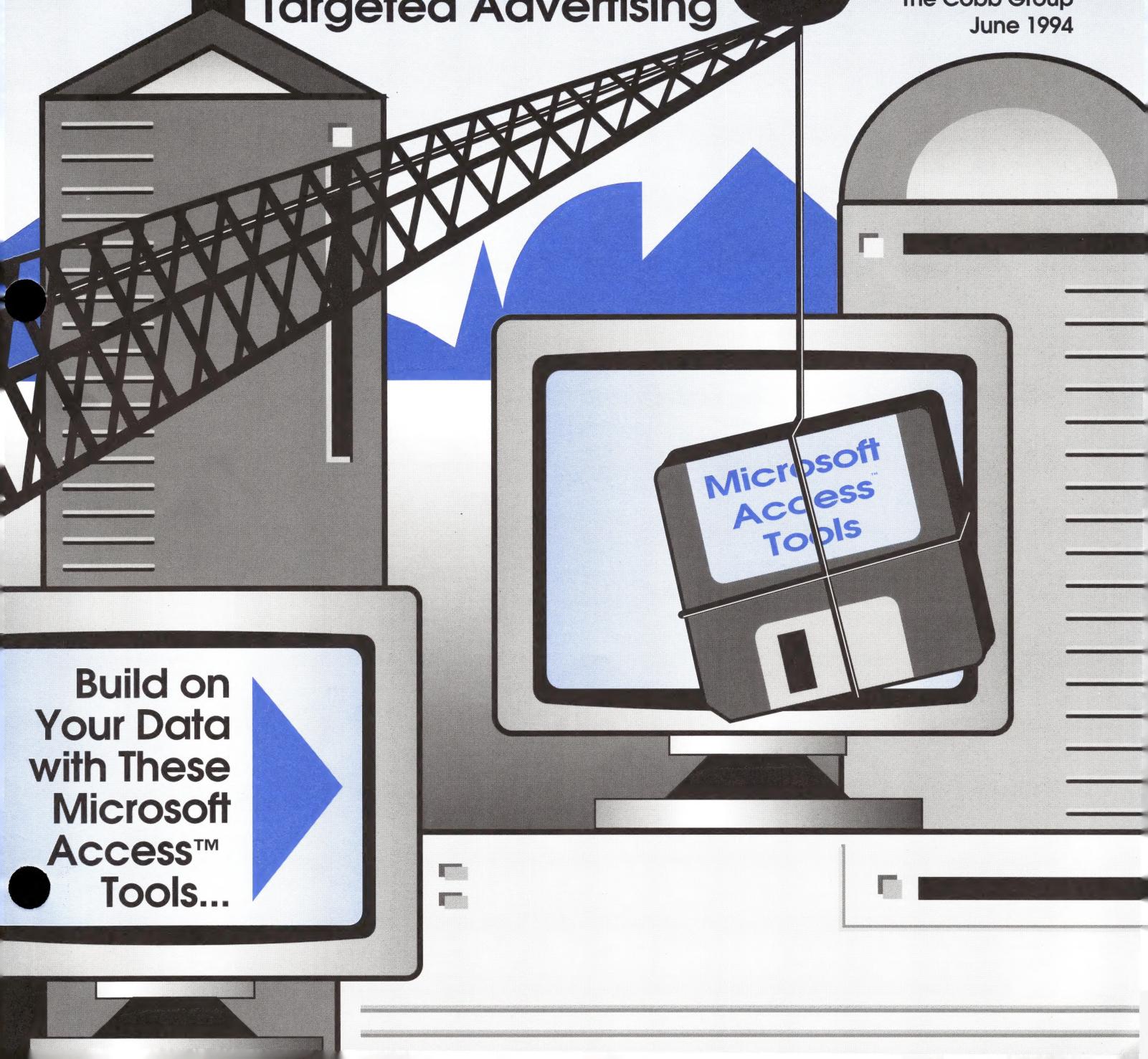
Now, double-click the Close box on the Order Review form to dismiss it. Next, click the Next Record button ( ) on the Employees form to advance to Andrew Fuller's employee record. Click the Order List button again. As Figure D shows, this time Access displays the records associated with Fuller. ❖

Microsoft Access™

spotlight

Targeted Advertising

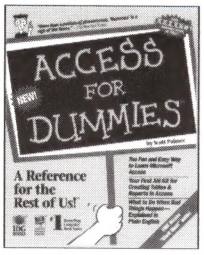
The Cobb Group
June 1994



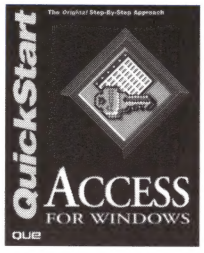
Build on Your Data with These Microsoft Access™ Tools...

SAVE 10-15% ON THE BEST ACCESS 2.0 BOOKS!

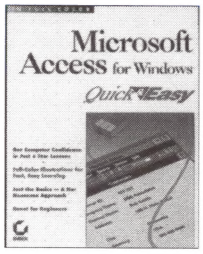
BOOKS AND VIDEOS FOR THE ABSOLUTE NOVICE



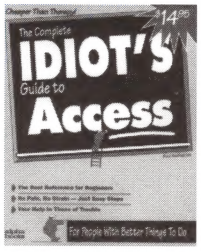
Access 2.0 for Dummies
Scott Palmer, 350 pgs.
$19.95, IDG Books, Easy to use, humorous, and simple for the absolute beginner!



Access 2.0 Quickstart
QUE, 350 Pages.
$21.95, QUE, Fast-paced, aggressive learning for mid-range Access 2.0 users

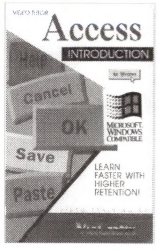


Access 2.0 Quick & Easy
Robert Cowart, 300 pgs.
$19.95, Sybex, Step-by-step instruction and color visuals for the beginning user.



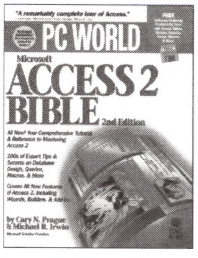
Idiot's Guide to Access 2.0
Paul McFedries, 300 pgs.
$14.95, Alpha, Humorous guide to databases in general and Access in particular.

3 Video Set

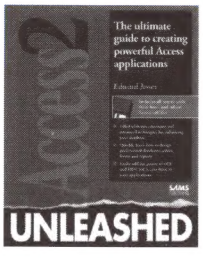


Access 2.0 Video Bundle*
Introduction, Intermediate, Advanced, **$139.95**
3 Videos from today's leader in video training!

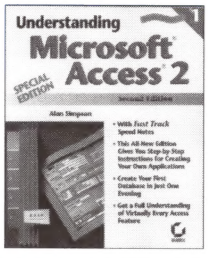
BOOKS FOR TABLES, FORMS, QUERIES, REPORTS, AND MACROS



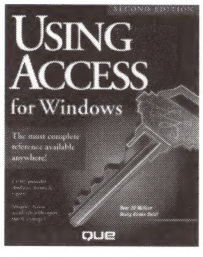
Access 2 Bible*
Prague/Irwin, 1000 pgs.
$39.95, IDG Books, The most comprehensive book for new Access 2.0 users.



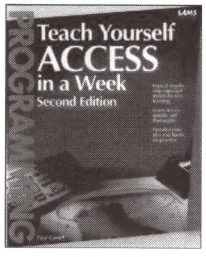
Access 2 Unleashed*
Ed Jones, 900 pgs.
$34.95, SAMS, Perfect for learning fundamentals plus some advanced topics.



Understanding Access 2.0
Alan Simpson, 1100 pgs.
$29.95, Sybex, For both end-users and developers with extensive coverage

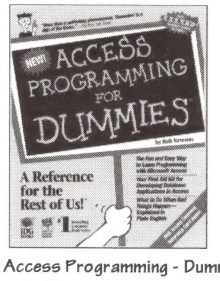


Using Access 2.0 for Windows
Roger Jennings, 1200 pgs.
$29.95, QUE, Combines step-by-step lessons with power-user techniques.

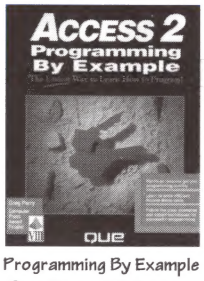


Teach Youself Access 2.0
Paul Cassel, 550 pgs.
$24.95, SAMS, Learn the essentials of Access 2.0 in only a week!

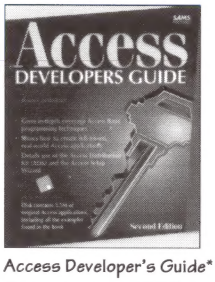
BOOKS TO LEARN ACCESS BASIC PROGRAMMING



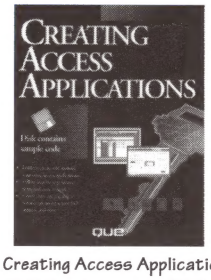
Access Programming - Dummies
Rob Krumm, 350 pgs.
$19.95, IDG Books, The ideal book for budding Access 2.0 programmers.



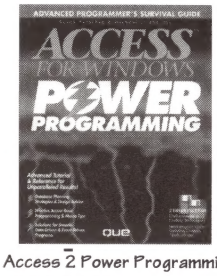
Programming By Example
Greg Perry, 500 pgs.
$29.95, QUE, Aimed at exploring Access 2 Basic for beginning programmers



Access Developer's Guide*
Roger Jennings, 1200 pgs.
$44.95, SAMS, Great coverage of all aspects of designing 2.0 applications.

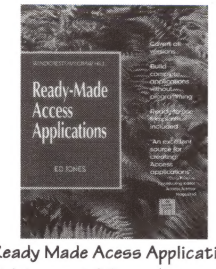


Creating Access Applications*
QUE Developement, 550 pgs.
$49.95, QUE, Covers from data entry to advanced reporting in Access 2.0

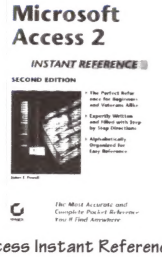


Access 2 Power Programming*
QUE,1,000 Pages
$49.95, QUE, Addresses the advanced Access 2 developer with BASIC code.

OTHER OFFERINGS



Ready Made Acess Applications*
Ed Jones, 400 pgs (Access 1.1)
$36.95, McGraw Hill, Not about programming but a great set of simple applications



Access Instant Reference
James Powell,, 300 pgs.
$9.99, Sybex, Tips, hints, and notes on all Access 2.0 features

US Phone Orders: (800) 277-3117

*Includes Disk With Book or Videos

Qty	Title	Price	Total

Buy 1 book save 10%!
Buy 2 or more save 15%!!

Subtotal: _____
Less Discount: _____ *
Shipping & Handling: _____
Total: _____

* **USA Orders: Add $ 3.95 per book**
* **Canadian Orders: Add $ 6.95 per book**
* **Europe/S. America: Add $25.00 per book**
* **Pacific Rim/Africa: Add $35.00 per book**

Call for Free 12 Page Detailed Catalog

Fax/International Orders
(203) 644-5891

Mail/Fax Orders: Fill in the Form Below and Send To:

Cary Prague *Books and Software*
60 Krawski Dr.
S. Windsor, CT 06074

Name_____
Company_____
Address_____
City_____State____Zip_____
Country_____Phone_____
Payment: ⌐Check ⌐Visa ⌐MasterCard ⌐Amex
Card#_____Exp. Date_____

All books shipped within 24 hours - 2nd Day Priority Mail
International Orders Shipped Air Mail Overnight Ship Available

5 NEW SOFTWARE PRODUCTS FOR ACCESS 2.0!

New Report Wizards That Work Great Together!

ENVELOPE REPORT WIZARD

The Envelope Report Wizard lets you create a report design for envelopes quickly and easily. Because it creates a standard Access report you can do almost anything with it!

- Choose from a Wide Variety of Envelope Sizes
- Customize the Completed Report Design
- Works Completely Within Access

Only $69.95

MAIL MERGE REPORT WIZARD

The Mail Merge Report Wizard complements the Access report writer wizards. Using word-wrap technology already in Access, eight wizard screens similar to Access' own Mailing Label wizard help you create form letters easily without Word 6.0.

- Works With the Access Runtime Version
- Easily Change the Font for the Report

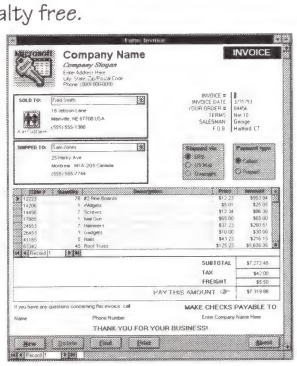
Only $79.95

Both for $129.95!

Access Business Forms Library

$79.95

The Access Business Forms Library is a collection of 35 professionally designed business forms that you can use in a standalone Access database container or integrate into your own Access Applications. You can distribute individual forms from the library royalty free.



Yes! I Can Run My Business With Access!

$199.95

You want to use Access to run your business. You want to create your own custom queries and reports, but --- you need a starting point.

Yes! I Can Run My Business With Access takes 5 basic business functions and automates them using simple Access tables, queries, forms, reports, and macros..

Reports Include:

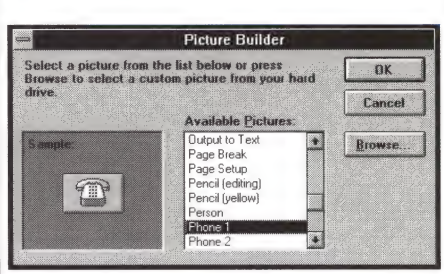
Customer Directory	Customer Credit
Invoice Summary	Invoice Detail
Daily Sales	Trend Graphs
Sales by Customer	Tax Summary
Receivables Aging	Inventory

Picture Builder Add-On Picture Pak

$99.95

We were the first company to offer picture buttons for Access 1.1. Now, for Access 2.0, we've done it again! Over 1500 fully customizable pictures, bitmaps, and icons for your command buttons, custom toolbars, and even toggle buttons. Icons include office, industry, and professions

Over 1500 Pictures



CALL FOR OUR 12-PAGE CATALOG!
See our other advertisement for software shipping costs!

US/Canada Orders (800) 277-3117
International and Fax (203) 644-5891

CARY PRAGUE BOOKS AND SOFTWARE - 60 Krawski Dr. S. Windsor, CT 06074

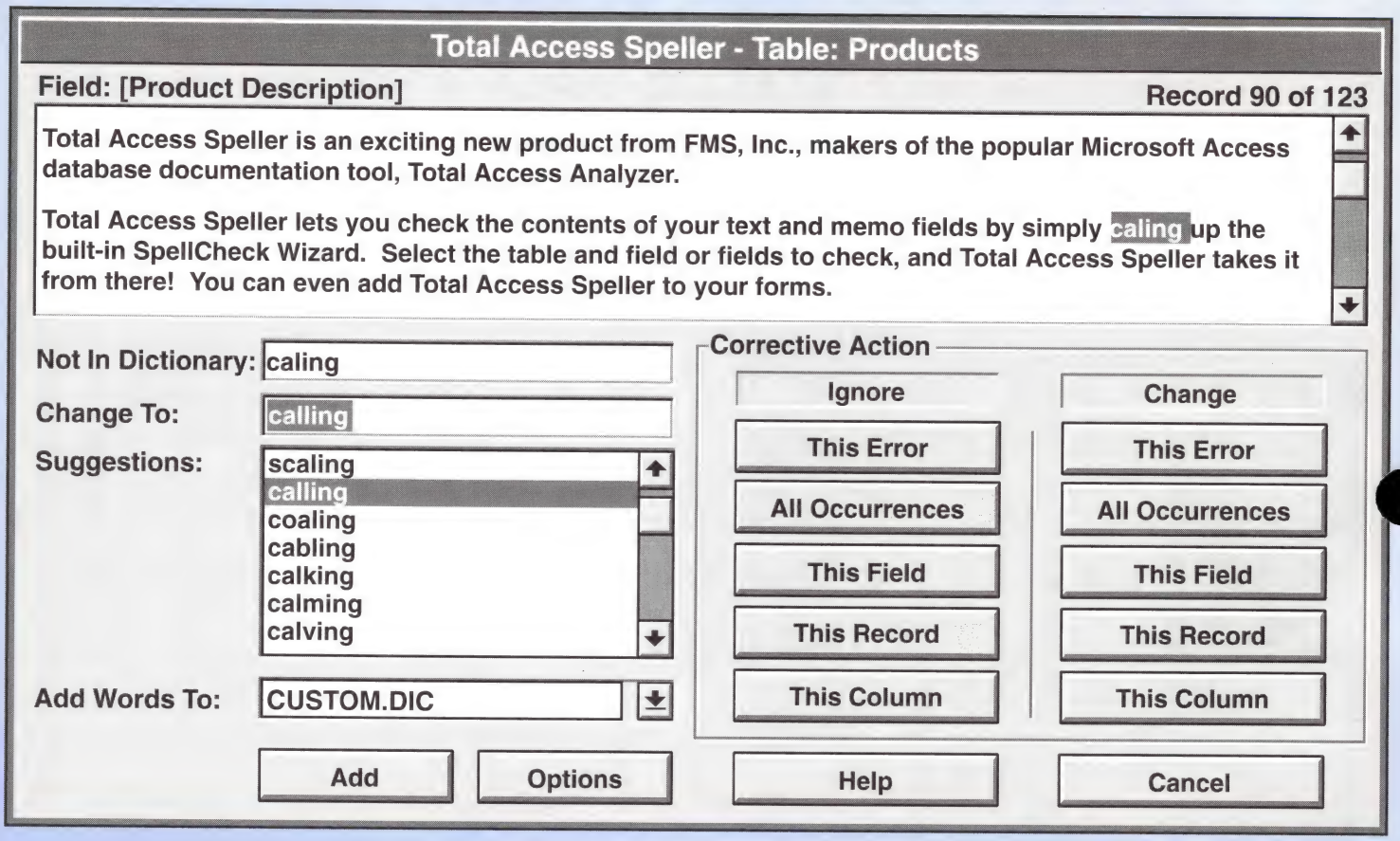
Total ACCESS Speller

Spell Checker for Microsoft Access® Databases

Supports Microsoft Access 1.1 and 2.0!

- Spell Check a Field or an Entire Table
- Check Text and Memo fields
- Runs from within Access
- 120,000+ Word Dictionary
- User Customizable Dictionary
- Multi-user Support
- Fast Performance

and more!



Total Access Speller is designed specifically for Microsoft Access and understands your tables even in multi-user situations. Spell check the current field, one field (column) for an entire table, or multiple fields in a table. Total Access Speller even works on queries!

The SpellCheck Wizard lets you quickly select the table and fields to check and walks you through the process. Features such as Ignore and Change are "table aware". In addition to handling the current error and all occurrences, Ignore and Change can be limited to just the current field, record, or column (current field for all records). Words not in the Total Access Speller's 120,000+ word dictionary can be added to your own custom dictionary.

Order Today!

Total Access Speller is available in a single user (network compatible) version, and a network 5-Pack which supports five simultaneous users and includes 5 user manuals. Call for additional user counts.

Take advantage of our special introductory price of only $69 for the single user version (that's $30 off!) or $199 for the network 5 pack. Add $5 S&H ($15 Int'l).

FMS, Inc.
8027 Leesburg Pike, Suite 410
Vienna, VA 22182 USA
(703) 356-4700, ext. 322
Fax: (703) 448-3861
Compuserve: 75160,3375

30 Day GUARANTEE Money Back

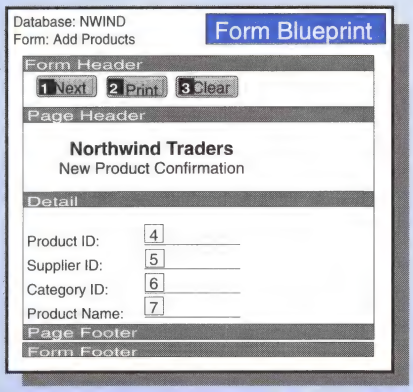
Total ACCESS Analyzer

Documentation and Analysis of Microsoft Access® Databases

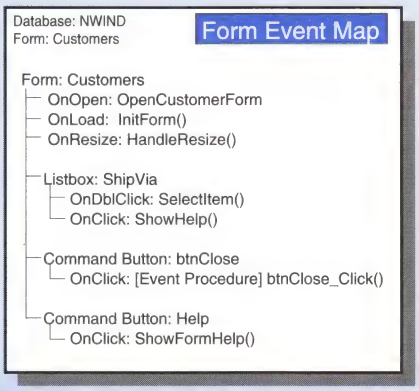
New Version for Microsoft Access 2.0!

- Module Documentation
- Security Documentation
- Procedure Flow Diagrams
- Variable Cross-Reference
- Separate Output Database
- 40+ New Reports
- Improved Performance

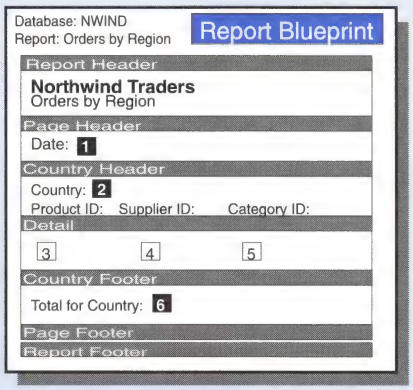
and more!



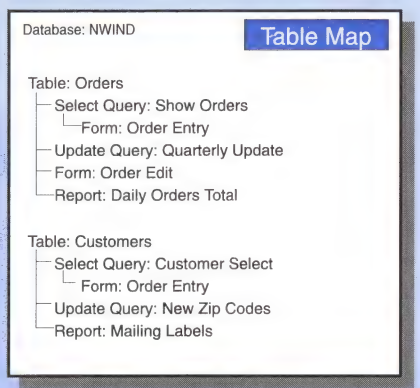
See visual representations of your forms with annotated controls. Black boxes highlight controls with attached code.



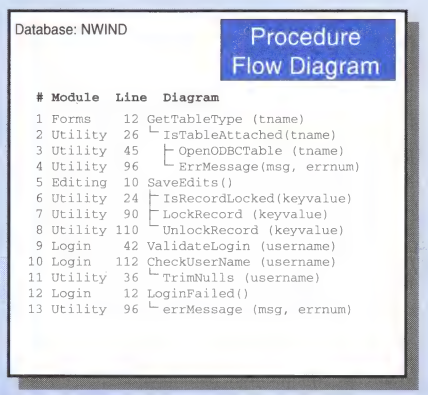
Quickly determine all the events on a form and its controls that call macros, procedures or event procedures.



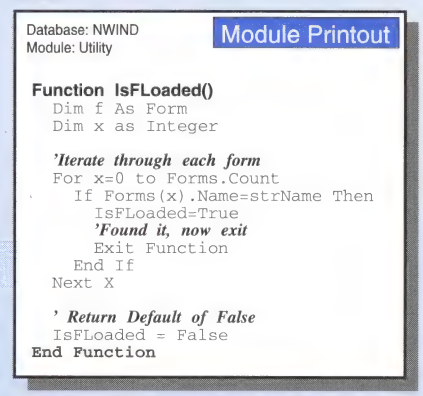
See visual representations of your reports with annotated controls. Black boxes highlight controls with attached code.



Quickly see how tables are used across queries, forms and reports. Diagrams also available for Forms and Reports.



See procedure calling chains even across modules! Modules, procedure names, and parameters are all shown.



Print module code with format options for procedure definitions, code, and comments. Specify bold, italics, fonts, and page headers and footers.

Complete Documentation!

- Tables
- Fields
- Indexes
- Queries
- Forms
- Reports
- Controls
- Properties
- Macros
- Modules
- Groups & Users
- Permissions
- Parent/Child Relationships
- Variable Cross-Reference
- Table, Field, and Query Cross Reference

and more!

Total Access Analyzer is designed specifically for Microsoft Access users. Best of all, the documentation is stored in Access tables. You have control of the results! Query the data, use our reports, modify them, or create your own. Developers, power users and database newcomers will all find Total Access Analyzer an invaluable aid in the design, implementation, and maintenance of Microsoft Access databases.

Order Today!

Take advantage of our special introductory price of only $145 (that's $50 off!). Upgrade from Total Access Version 1 for only $79. Add $5 S&H ($15 Int'l). Available in July.

VISA MasterCard

Total Access Version 1 still available for Access 1.1

30 Day Money Back GUARANTEE

FMS, Inc.
8027 Leesburg Pike, Suite 410
Vienna, VA 22182 USA
(703) 356-4700, ext. 322
Fax: (703) 448-3861
Compuserve: 75160,3375

Advertiser's Index

Microsoft Access SPOTLIGHT is published by The Cobb Group. For advertising rates and information, contact Tracee Bell Troutt (ext. 430) at (800) 223–8720.

"The Easiest, Least Expensive way to manage your paper documents."

Introducing PaperBridge™, the award-winning document imaging software rated "best value" by customers.

Fast, affordable and easy to use, PaperBridge for Access™ is such a life-saver, you'll wonder how you ever got along without it. It's like having your own private assistant. Purchase orders, invoices, resumes, legal documents, insurance claims – whatever documents you have – are now at your fingertips.

Here's how it works:

Simply copy your document to your PC using your scanner or fax — and let PaperBridge do the rest! It automatically displays the document on your computer screen and lets you . . .

- File your document in your database for easy and flexible retrieval
- Fax it
- Or use your OCR to take it into a spreadsheet or word processor

Plus, PaperBridge uses a standard desktop management metaphor that lets you handle your electronic documents like you would any paper document. You can move them around, zoom in on the parts you're interested in, highlight key points, write notations, and underline or blackout words or sections. You can even attach a note.

Automatic compression saves you disk space — and time.

A major software miracle, PaperBridge save you tons of disk space with the best compression algorithm in the business. Even large, cumbersome files are instantly turned into compact and efficient data files.

Award-winning PaperBridge receives critical acclaim.

The experts agree. PaperBridge is a must-have tool for state-of-the-art office management. That's why InfoWorld honored it with their prestigious Buyer Assurance Seal, and why Imaging Magazine elected this revolutionary software Product of the Year.

PAPERBRIDGE
BUYER'S ASSURANCE SEAL
INFO WORLD
September 1993

IMAGING MAGAZINE PRODUCT OF THE YEAR 93

Take a look at what early fans have to say about PaperBridge:

"PaperBridge brings unique strengths to linking images to Access applications. ...Even if you are not an Access expert, PaperBridge is surprisingly easy to use."

– Herb Edelstein & David Black, Review Board InfoWorld, September 1993

"PaperBridge provides an elegant way to create image-enabled applications without having to resort to kludgy tricks. The easiness and no coding approach of the PaperBridge Wizard makes this a tool that is usable by database developers from beginner to expert. You may spend the purchase price many times over trying to do yourself what PaperBridge does so quickly and easily."

— Michael Harding, Access Advisor, January 1994

PaperBridge works with Microsoft's Access to give you the best in document imaging- easier and cheaper than all the rest.

PaperBridge sticks to the stuff it knows best — image processing — in an open, non-proprietary approach. Instead of reinventing the wheel, Microsoft Access was selected for data management tasks. The result? You get a lower price and the easiest-to-use document management program ever released.

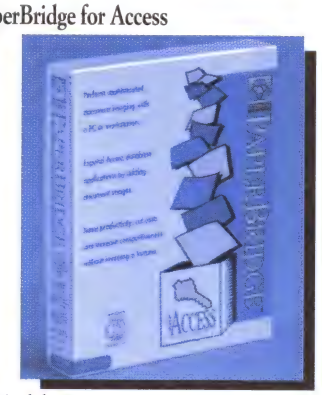
Satisfaction Guaranteed!

Evaluate PaperBridge with absolutely no risk for 60 days. Put it through its paces. If you are not happy for any reason, just send it back for a no-hassle refund.

Priced lower than you'd ever imagine.

Dollar for dollar, PaperBridge for Access is the best imaging buy on the market. You will be amazed that such a great product costs so little.

PaperBridge for Access



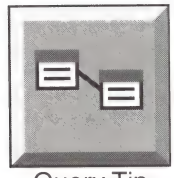
Don't delay -- act now!

For more information about how PaperBridge can solve your imaging needs,

Call 800-532-3198

or write to: **TEAMWorks/Optika**
65 Boston Post Road West
Marlboro, MA 01752

Supplying fixed column headings in a crosstab query



When you build a crosstab query definition, Access requires you to choose the Column Heading option in the Crosstab row for at least one field or expression. You must also select one field or expression for the row headings, and one field or expression for the values displayed in the query. However, the values returned by the field or expression that generates the column headings may not include all the columns you want in your query. Depending on the type of value Access uses for your column headings, the columns in your query may not appear in the order you want.

For example, if you use month names for the column headings, Access will display columns only for months in which the query returns at least one value. Consequently, if a query returns no information for the months of March, June, and October, Access will display the query results without showing columns for those months. However, you can use the Column Headings property to provide a fixed list of column headings. In this article, we'll show you how to define fixed column headings for a crosstab query.

Choosing column headings

Access provides several levels of control over column headings. First, you can accept the default format returned by the field or expression you use to designate the column headings. Typically, this will be a date, numeric value, or text label. Sometimes, you may want to use a formatting function to control the appearance of the column headings. For example, you may want to restrict a date heading to month names or to convert a text heading to uppercase.

Although entering fields and expressions on the QBE grid provides a certain level of control over the column headings, you may need to customize the column headings a bit further. Access allows you to supply a list of column headings in the Query Properties dialog box. Although Access 1.1 and Access 2.0 display the Query Properties dialog box differently, the Column Headings (Fixed Column Headings in Access 1.1) property works similarly in both versions.

The Column Headings property allows you to type a list of headings for Access to use when it displays your query. These headings must match the data returned by the field or expression that generates the column headings for the query. If you supply a value in the Column Headings property that doesn't appear among the values returned by the field or expression, Access places that column heading over an empty column. Therefore, you can use fixed column headings to hold a place for columns that contain no data.

The Column Headings property also governs the order in which the columns appear in the query. Now, let's build a crosstab query that uses fixed column headings.

Building the query

We'll use the Orders table in the database NWIND.MDB to build our query. Our query will display the number of orders registered each month by each country during a particular year. We'll use the Column Headings property to hold places for months in which no orders shipped.

Begin by opening the NWIND.MDB database. Then, select the Orders table from the Database window. Click the New Query button () on the toolbar to display the New Query dialog box. Then, click the New Query button to open the QBE grid. (If you're using Access 1.1, you'll see the Select Query dialog box, which contains the QBE grid.)

Next, choose Crosstab from the Query menu or click the Crosstab Query button () on the toolbar. Drag the Ship Country field into the first column of the grid. Click on the Crosstab row, and then choose Row Heading from the dropdown menu.

Now, drag the Order Date field into the second column of the grid. Click the Crosstab row and choose Column Heading from the dropdown menu.

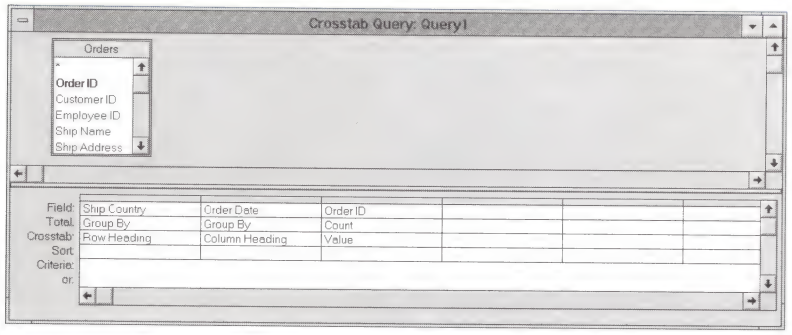
Drag the Order ID field into the third column of the grid. Click on the Total row of the third column and choose Count from the dropdown menu. Next, click on the Crosstab row and choose Value from

the dropdown menu. At this point, your query definition should look like the one shown in Figure A.

This query should display a series of date values as the column headings. The row headings should list the countries that received shipments. The table will display the number of products shipped to each country. Click the Datasheet View button (▦) on the toolbar to display the result of the query.

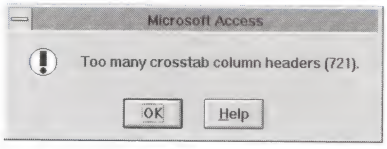
After a few seconds, Access will display the error message shown in Figure B, indicating that the query requires too many column headers. Since we accepted the default format for the Order Date field, Access tried to generate a separate column for each day during the period covered by the Orders table.

Figure A



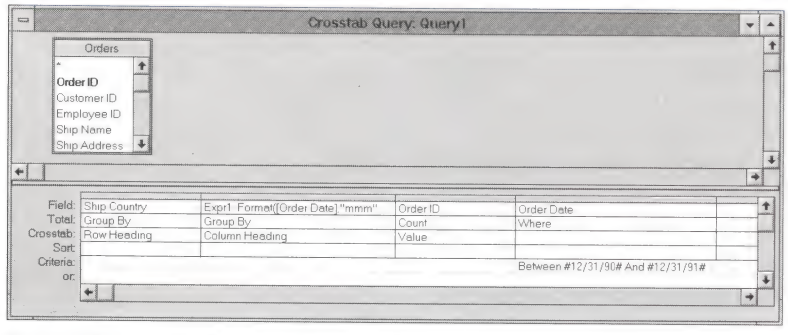
Your crosstab query definition should look like this.

Figure B



Access generates this error message when you try to execute the query.

Figure C



Make these changes to your query definition.

Restricting the dates

To solve this problem, let's alter the query so that it displays columns by month instead of by day. First, select the cell that contains the field name Order Date. Then, type the expression

```
Format([Order Date],"mmm")
```

Since the table's information spans more than one year, let's restrict the query to a single year. To do so, we'll add another column to the query, but we'll tell Access not to display it as part of the query results. We'll use this column to govern the dates the query covers.

To create the column, drag the Order Date field into the fourth column of the grid. Then, click on the Total row and select Where from the dropdown menu. In the Criteria cell, type

```
Between 12/31/90 and 12/31/91
```

(If you're using Access 1.1, enter *Between 12/31/89 and 12/31/90.*) At this point, your query definition should look like the one shown in Figure C.

Now, click the Datasheet View button on the toolbar to display the new query result. As Figure D shows, the query displays abbreviations for the month names at the top of each column and presents the information we want. However, the columns appear in alphabetical instead of chronological order. Also, the query doesn't include columns for the months January through April.

Since the Format() function returns a string value instead of a date value, Access determines that the most reasonable way to organize the values is alphabetically. The columns for January through April are missing because they contain no data for the calendar year 1991. You can solve both these problems by adding fixed column headings to your query.

Fixing the column headings

To set up fixed column headings, first click the Design View button (▦) on the toolbar to return to Design View. As we mentioned earlier, although Access 1.1 and

Access 2.0 handle fixed column headings similarly, there are a few differences. We'll point out those differences as we work though this example.

If you're using Access 2.0, choose the Properties button () from the toolbar to display the Properties window. If the Properties window doesn't display the title Query Properties, click anywhere on the QBE grid to display the query properties in the Properties window. You'll enter the list of column headings in the Column Headings text box.

In Access 1.1, you simply choose Query Properties... from the View menu to display the Query Properties dialog box. Click Fixed Column Headings to activate the Fixed Column Headings feature.

You enter the column headings in the same format in both Access 1.1 and Access 2.0. Type the following column heading definition into the location appropriate for the version of Access you're using, either Column Headings or Fixed Column Headings:

```
"Jan","Feb","Mar","Apr","May","Jun",
"Jul","Aug","Sep","Oct","Nov","Dec"
```

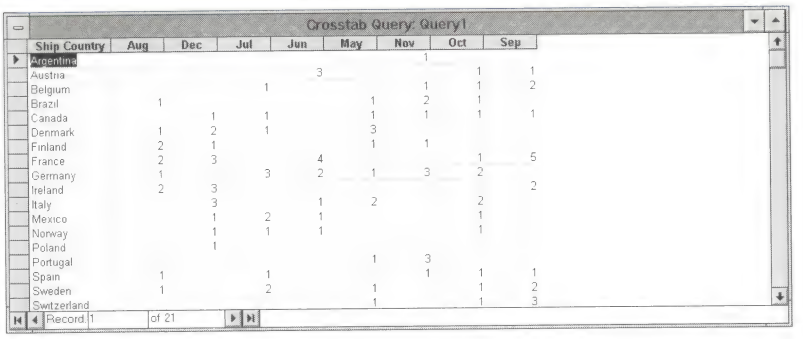
If you're using Access 1.1, click OK to dismiss the Query Properties dialog box when you finish. Figure E shows the completed Column Headings definition in the Access 2.0 Query Properties window.

Now, click the Datasheet View button on the toolbar to display the query result. As Figure F shows, the query now contains column headings for each month of the year. The query also displays the months in the order you established with the Column Headings property.

Notes

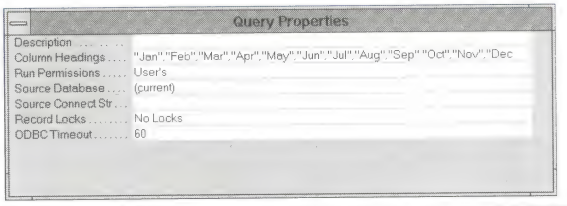
When you supply fixed column headings, make sure you provide a heading for each column you want to display. Access will display only the columns in which the value returned by the field or expression that generates the column headings matches an item in the column heading list you supply. For example, if you exclude Oct and Nov from the list of headings we gave the Column Headings property, Access will eliminate those columns from the query—even if they contain data.

Figure D



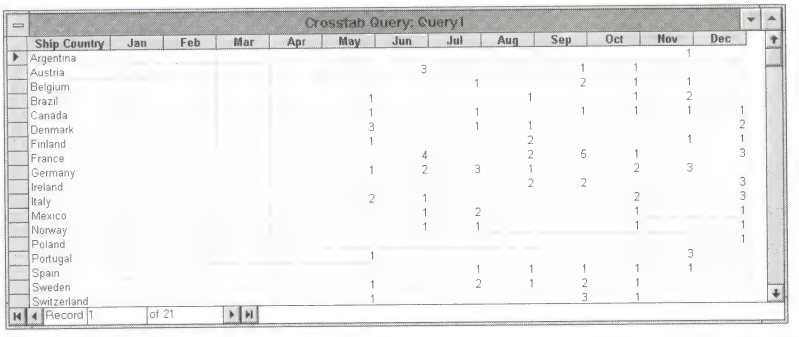
Although this query returns the information you want, it's difficult to read.

Figure E



The Column Headings text box looks like this in Access 2.0.

Figure F

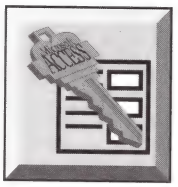


The Column Headings property reorganized the columns to make the query easier to read.

Also, the values you give the Column Headings property must exactly match the ones returned by the field or expression. For instance, if the expression returns March and you supply the Column Heading property with the value Mar, Access will display an empty column titled Mar, but won't display a March column.

Conclusion

Crosstab queries are a great way to present summary data related to two fields in a table. The Column Headings property lets you supply a list of column headings to make a crosstab query more readable. ❖



Making Access easier on your eyes

Microsoft Access offers a number of tools designed to help you work faster. The toolbar lets you execute many Access commands by simply clicking an icon. You can automatically move from Design View, Form View, or Query View to Datasheet View when you need to see information from several records at once. By default, Access displays toolbar icons and datasheet text at a fairly small size so that you can see more information onscreen without resizing your window or scrolling through the datasheet.

However, if you use your computer for several hours every day, you may experience eye strain. This can be especially problematic if you wear corrective lenses. Fortunately, Access lets you modify the default size settings for toolbar icons and datasheet text. In this article, we'll show you how to modify these default settings to make Access easier on your eyes.

Making the changes

It's easy to modify the default size settings for toolbar icons and datasheet text. You use the same steps to establish default datasheet text settings in Access 1.1 and Access 2.0. In both versions, you simply open the Options dialog box and establish the necessary settings. Unfortunately, in Access 1.1, you can't change the toolbar icon size. Now, let's begin by changing the toolbar icon size in Access 2.0.

Altering the toolbar

If you run Windows in 640x480 mode, you probably won't need to change the default size of the toolbar icons. However, in higher-resolution modes, the toolbar icons appear quite tiny. This is particularly acute in 1024x768 video mode. (We discuss Windows' video modes in "Changing Windows' Video Resolution" on page 14.) We'll show you two strategies for manipulating toolbar icon size—one for high-resolution settings and one for low-resolution settings.

High-resolution video

If you're running in 1024x768 mode, you'll definitely want to increase the toolbar icon size. Even if you have excellent vision, it's easy to confuse some of the more complex icon designs at their default size, shown in Figure A. Let's begin by increasing the toolbar icon size while we're operating Access in high-resolution video mode.

To open the Options dialog box, first open any database. Then, choose Options... from the View menu. The Options dialog box lets you establish default settings for several categories of appearance and action properties. The property for toolbar icon size appears in the General category.

First, select General from the Category list box. Then, scroll through the list until you locate the Large Toolbar Buttons property. Change the default setting from No to *Yes*, as shown in Figure B.

When you finish, click OK to dismiss the Options dialog box. Access will adjust the toolbar icons to a more reasonable size, as shown in Figure C.

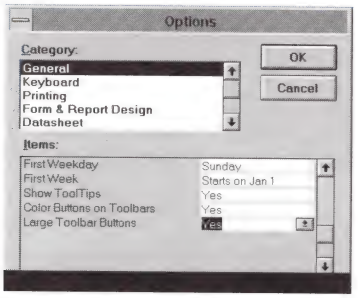
Low-resolution video

If you use Windows in 640x480 or 800x600 mode, Access won't be able to display large toolbar icons below the main title bar.

Figure A

In a high-resolution video mode, toolbar icons are too small.

Figure B



Change the Large Toolbar Buttons property to Yes.

Figure C

You can now identify the toolbar icons more easily.

The toolbar becomes too large, and the icons disappear beyond the right margin of the screen. However, if your goal is to increase the icon size as much as possible, you may want to use large toolbar icons with a low-resolution video setting.

If you use Access 2.0, you can display the toolbar as a palette to show all the large toolbar icons in low-resolution video. To do so, first set the Large Toolbar Buttons property to Yes, as we described earlier. Then, place the tip of the mouse pointer on an area of the toolbar not occupied by an icon.

Hold down the mouse button and drag the toolbar into the main work area. As Figure D shows, the toolbar becomes a free-floating palette. Although it obscures some of the work area, you can see all the icons in their largest available size. To return the toolbar to its default position, drag the toolbar palette to the top of the window and release the mouse button. Now, let's modify the default text size for Datasheet View.

Changing the datasheet text size

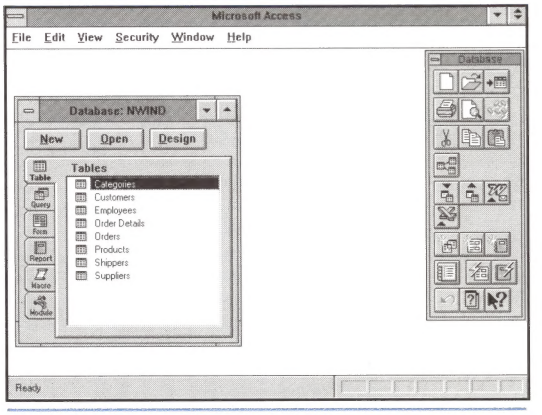
By default, Access chooses a default datasheet text size setting of 8 points. Although Access can display more data in 8-point type, text that small may be difficult to read. The Options dialog box lets you choose a more readable default text size.

To increase the datasheet text size, open any database and choose Options... from the View menu. Then, select Datasheet from the Category list box. Next, change the Default Font Size setting from 8 to *12*, as shown in Figure E. Click OK to dismiss the Options dialog box.

Now, when you click the Datasheet View button to switch to Datasheet View from any other view, Access will display your text in 12-point type. Access will also use this setting for tables when you double-click the table name in the Database window.

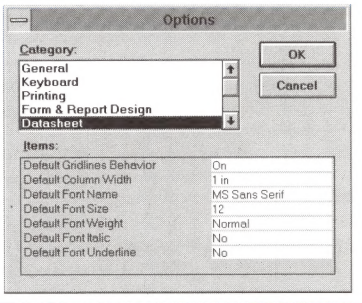
When you build a form, table, query, or report, you can establish text size settings for that document. For example, you could set the text size of a query's results to 12 points. When you save that query, Access will also save the text size setting. Access will honor that setting regardless of the default datasheet text size setting.

Figure D



If you display the toolbar as a palette, you can see all the icons at their largest size in low-resolution video mode.

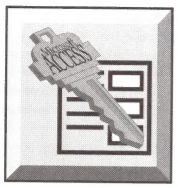
Figure E



You can increase the default datasheet text size in the Options dialog box.

Conclusion

Access automatically selects default sizes for datasheet text and toolbar buttons. However, you can establish new default settings for your datasheet text size. In Access 2.0, you can also change the default size settings for toolbar buttons. ❖



Changing Windows' video resolution

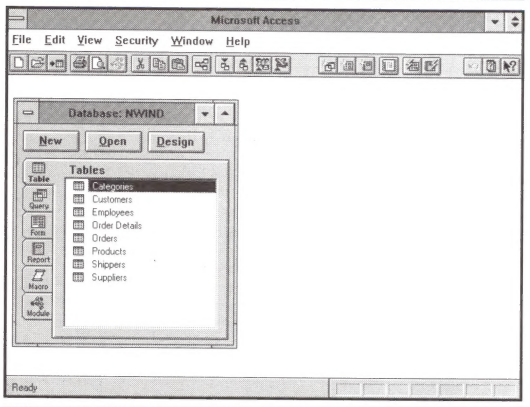
Microsoft Access takes a modular approach to database design. You create individual tables for related information. You build queries, forms, and reports to display and manipulate the data in the tables. You can create macros and Access basic modules to customize your databases. With all these components, you may quickly run out of screen space in the Access work area.

Although a number of manufacturers offer oversized monitors for your PC, you may be able to expand your current monitor's real estate by making a simple change in Windows Setup. In this article, we'll show you how to choose a high-resolution video setting to increase the working area on your monitor.

Getting ready

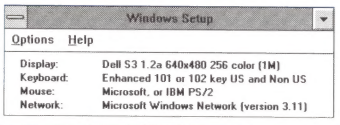
When you installed Windows, you probably accepted the default video settings along with the other default settings Windows established. In a normal installation, Windows chooses a 640x480 video resolution. Figure A shows Microsoft Access maximized in 640x480 video mode.

Figure A



Access looks like this in 640x480 video mode.

Figure B



The Windows Setup dialog box shows your current Windows display settings.

However, many monitors are capable of much higher video resolution. Typically, you can select 800x600 or 1024x768 modes with SVGA monitors.

The exact video resolution you can use depends on your system hardware and the video drivers that came with your system. Even if your hardware supports high-resolution video, you'll need the proper video driver to tell Windows how to interact with your system.

Choosing the correct driver

Every video adapter requires its own video driver. The driver usually includes the manufacturer's name or a description of the monitor type in the setting name. For example, if you have a SVGA video adapter manufactured by Dell, you should look for the designation Dell or SVGA in the Display dropdown list in the Windows Setup Change System Settings dialog box. Most manufacturers (OEMs) include diskettes that contain the correct video drivers with their products.

The video drivers may have been installed on your hard drive before you purchased your system. If that's the case, you won't have to prepare anything before you increase your video resolution.

If you set up your own system, you probably copied the video drivers from a set of diskettes supplied with your monitor. Unless you copied all the drivers to your hard drive before you installed Windows, you may need to locate the diskette with the correct high-resolution video driver before you enter Windows Setup. Now, let's use Windows Setup to increase your video resolution from 640x480 to 1024x768.

Using Windows Setup

First open the Main program group on your Windows desktop. Next, double-click the Windows Setup icon to open the Windows Setup dialog box. As Figure B shows, the dialog box lists your current display settings.

Now, choose Change System Settings... from the Options menu. Click the drop-down arrow beside the Display text box to see the installed video drivers. Choose the correct 1024x768 driver for your display. We selected

```
Dell S3 1.2a 1024x768 256 color
    (IM Large fonts)
```

as shown in Figure C.

If you need to install a driver from your original diskettes, scroll to the end of the list and select

Other Display (Requires disk from OEM)

After you make your selection, click OK.

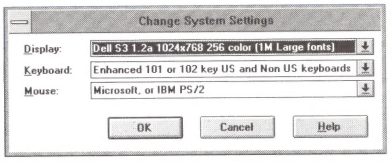
If you selected the Other Display option, Windows will prompt you for the OEM diskette. Insert the diskette into drive A and follow the onscreen instructions to install the correct driver.

After you select the correct driver, Windows will ask if you want to use the driver currently installed on your system. Click Current unless you want to use a diskette to update that specific driver.

Finally, Windows will need to restart your system before the change can take effect. Click the Restart Windows button to complete the video setup. When you return to Windows, the desktop should open in high-resolution video. The text and icons will appear smaller, but you'll have significantly more space available on your screen. Figure D shows Microsoft Access maximized in 1024x768 mode.

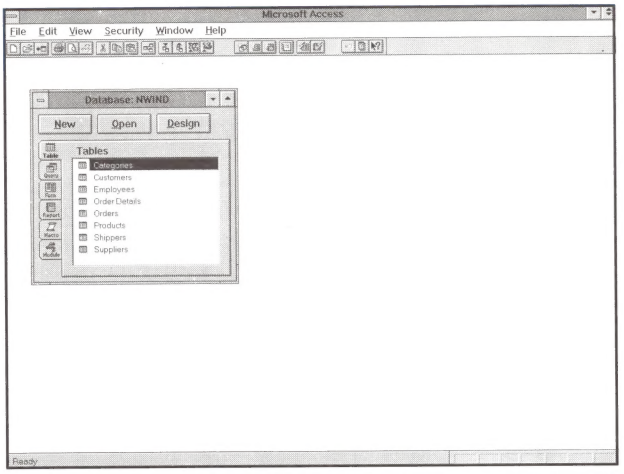
After you select a high-resolution video mode, you may want to modify some of the default display settings in Access. See "Making Access Easier on Your Eyes" on page 12 for more information. You may need to use trial and error to determine which settings work best for you. If you can't locate the correct driver for your system, contact your system's manufacturer. If you installed a third-party video adapter, contact its manufacturer.

Figure C



Choose the correct video driver for your system.

Figure D

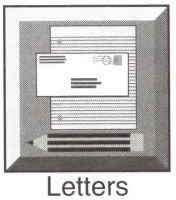


You have more working space in 1024x768 mode.

Conclusion

When you open several components of an Access database, you may run out of screen space. You can increase your available screen space by selecting a higher resolution video driver in Windows Setup. ❖

Finding the correct path in MSACCESS.INI


Letters

I recently purchased a copy of Microsoft Office. Since I already own Access, I didn't get the copy of Office that includes Access. However, I decided to move Access on my hard disk from the location in which I originally installed it to the MSOFFICE subdirectory. File Manager let me do this easily enough. I simply dragged the ACCESS subdirectory into the MSOFFICE subdirectory.

Unfortunately, after I did so, I couldn't start Access. Windows presented me with an error message saying that the path to the SYSTEM.MDA file was invalid. How can I correct this problem?

Georgia Sellars
Brownsboro, Vermont

Although File Manager can save you a lot of time when you need to move an entire subdirectory, it can't anticipate the individual needs of the applications installed on your system. When you make a change like the one Ms. Sellars describes, you must make a few adjustments manually. In this case, you need to change the path to certain critical files in MSACCESS.INI.

What is MSACCESS.INI?

The MSACCESS.INI file includes a number of path statements that tell Windows and other programs exactly how to locate Access. When Ms. Sellars moved Access on her system, the path statements in

SECOND CLASS MAIL

Microsoft Access
Technical Support
(206) 635-7050

Please include account number from label with any correspondence.

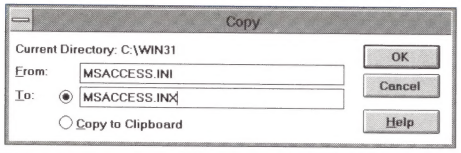
MSACCESS.INI became invalid. To fix the problem, you simply open MSACCESS.INI with Windows Notepad and change the path statements to reflect Access's new location. To demonstrate, let's work through a simple example.

Updating MSACCESS.INI

Suppose you originally installed Access in the APPS subdirectory of your system. Then, like Ms. Sellars, you decide to move Access to the MSOFFICE subdirectory. First, you use File Manager to move the Access subdirectory. Then, you open MSACCESS.INI and make the appropriate changes. As always, you should make a backup copy of any critical system files before you modify them.

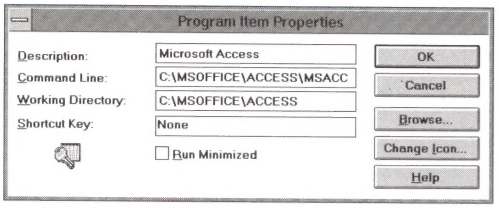
Begin by opening File Manager. Then, navigate to the APPS subdirectory. Click on the folder icon to display the subdirectories in APPS. Next, drag the ACCESS subdirectory icon onto the MSOFFICE subdirectory icon. When you release the mouse button, Windows will display a confirmation message box as it moves the ACCESS subdirectory.

Figure A



Make a backup copy of MSACCESS.INI before you alter it.

Figure B



You also need to change the settings in the Program Item Properties dialog box.

Now, double-click the ACCESS subdirectory icon to open it. Locate the file MSACCESS.EXE and double-click it to start Access. Instead of loading Access, Windows presents an error message telling you that the path

```
C:\APPS\ACCESS\SYSTEM.MDA
```

is invalid.

To update MSACCESS.INI, double-click the WIN31 subdirectory icon to open it. First, we'll make a backup copy of MSACCESS.INI. Select the file MSACCESS.INI. Then, choose the Copy... command from the File menu or press [F8]. Type *MSACCESS.INX* in the To text box, as shown in Figure A. When you click OK, Windows makes a copy of MSACCESS.INI named MSACCESS.INX. Windows stores this file in the WIN31 subdirectory.

Now, double-click the file MSACCESS.INI to open it in Windows Notepad. Next, scroll through the file to locate references to the old path to the ACCESS directory. Change any references from

```
C:\APPS\ACCESS\
```

to

```
C:\MSOFFICE\ACCESS\
```

You'll find references to the old path in the [Options] section, as well as in some of the database filters.

When you finish changing the statements, choose Save from the File menu. Then, choose Exit from the File menu to close Notepad. You should now be able to launch Access from File Manager with no problems.

Updating the Windows program item

You'll also need to modify the Access program item. First, locate the Microsoft Access program item on your desktop. Click on the program item to select it. Then, press [Alt][Enter] or choose Properties from the File menu to display the Program Item Properties dialog box.

Next, change the path statements in the Command Line and Working Directory text boxes to reflect the new path, as shown in Figure B. Click OK to save the changes and dismiss the dialog box. Now, you can start Access by double-clicking the Microsoft Access program item. ❖

Printed in the USA
This journal is printed on recycled and recyclable paper.
50% recycled fiber including 10% post-consumer waste